Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

A personalized TV Guide System

An Approach to Interactive Digital Television

Paulo Muniz de Ávila
Department of Computer Science - DC
Federal University of Sao Carlos - UFSCar
Sao Carlos, Brazil
paulo_avila@dc.ufscar.br

Sergio Donizetti Zorzo
Department of Computer Science - DC
Federal University of Sao Carlos - UFSCar
Sao Carlos, Brazil
Zorzo@dc.ufscar.br

***Abstract*— An essential change has been occurring in TV nowadays in Brazil: the migration from the analogical system to the system digital TV. A major impact of this change is the increasing capacity transmission of new channels. The Electronic Programming Guide helps viewers to navigate between channels, but the measure that new channels are available it is inevitable that information overload occurs making systems EPG inadequate. This situation arises the need of personalized recommendation systems. In this paper we present a recommendation system compliant with the middleware Ginga. Are presented the results obtained with three different mining algorithms running in a set-top Box using real data provide by IBOPE Midia. The IBOPE Midia is the company of the IBOPE group responsible for the communication, media, consumption and audience research.**

***Keywords*— Personalization, Multimedia, Recommendation System, Digital TV.**

I. INTRODUCTION

An essential change has been occurring in TV nowadays in Brazil: the migration from the analogical system to the system digital TV. This change has two main implications: the increase in the transmission of new channels with the same bandwidth and the possibility of sending multiplexed applications with the audio-visual content. As new channels emerge due to the transmission increase, it is necessary to create ways that allow the TV viewers to search among these channels.

The Electronic Program Guide (EPG) helps the TV viewers. However, as new channels are available, an information overload is unavoidable making the EPG system inappropriate. In Shangai [1], a big city in China, the TV operators provide different services (in the analogical system, channels), and this number has been increasing at a rate of 20% per year. This way, the traditional EPG system became unattractive because it takes too long for the viewers to search in the hundreds of options available to find their favorite program. In face of this situation, the personalized recommendation systems are necessary. Different from the EPG functions which allow basic search, a personalized TV system can create a profile for each TV viewer and recommend programs that best match this profile, avoiding the searching in many EPG options to find the favorite program. The TV viewer's profile can be realized in an explicit way where the system receives information about the preferences or it can be realized in an implicit way where the system can infer the TV viewers' preferences analyzing their behaviour background. In the DTV context, the implicit option is surely the best in face of the limitations imposed by the remote control to the data income. However, both systems can be used. To make the benefits (new channels, interactive applications) offered by the digital system possible, the TV viewers with analogical system need a new equipment called set-top Box. Set-top Box is a device which works connected to the TV and converts the digital sign received from the provider to audio/video that the analogical TV can exhibit. To have the advantages offered by the digital TV, the set-top Box needs a software layer which connects the hardware to the interactive applications called middleware. The DTV Brazilian System middleware is Ginga [2,3].

This paper proposes an extension to Ginga-NCL middleware through implementation of a new module incorporated to the Ginga Common Core called Recommender. The Recommender module is responsible for gather, store, process and recommend TV programs to the TV viewer. To develop the Recommender module, it was used the Ginga-NCL middleware developed by PUC-RIO (Pontifical Catholic University of Rio de Janeiro), implemented in C/C++ language with source code available under GPLv2 license and according with the patterns defined by the Brazilian system digital television [4].

The rest of this paper is organized as follow: section 2 presents related works, section 3 describes the providers, section 4 presents a general view of Ginga-NCL middleware and the extensions proposed to support the recommendation system; section 5 presents details of implementation and connection of new modules to Ginga-NCL middleware; section 6 details the experiences, the simulation environment and the results and section 7 presents the conclusion

II. RELATED WORKS

In the last years, many TV personalized systems have been developed to help the viewers in face of the increasing offer of new services. The first recommendation systems used explicit approach to register the viewer's preferences. In the last years, researches have been done aiming to infer automatically the viewer's preferences.

TV-Advisor [5] uses explicit techniques to create recommendation, demanding the viewer to specify the interests to the recommendation system.

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

PTV [6] is a system which provides personalized recommendation to the viewer based on the collaborative filtering approach. The viewer's preferences are recorded in an explicit way.

The Multi-Agent TV Recommender [7] matches both explicit and implicit methods to store the viewer's preferences and uses collaborative filtering in the recommendation of TV programs.

In [8] a multi-agent architecture for an adaptable EPG system is presented. The viewer's preferences depend on the day and time he/she watches TV. The viewer's profile is generated using explicit and implicit techniques.

AIMED [9] proposes a recommendation mechanism that considers tendencies like: mood, demographic information, etc, to recommend programs. The approach used is the hybrid approach based on the content filtering methods and the collaborative filtering.

The Recommender TV system uses implicit techniques to have the viewer's profile. Implicit techniques demand monitoring and analysis of the viewer's behavior background to have the profile.

III. SERVICE PROVIDER

This section presents important concepts related to the service provider, how the digital sign transmission is done and what information is provided and the relation with the recommendation system proposed in this paper.

Besides the transmission of audio and video, the Brazilian system digital TV is supposed to send data to the TV viewer. The service providers can send via broadcast application written in Java^{TM} known as Xlets or NCL applications, and both of them are defined in the television Brazilian system. Besides the application, the providers send tables which transport information to the set-top Box. This section gives details about two important tables to this context, the EIT (Event Information Table) and the SDT (Service description Table).

Open digital TV systems adopt the pattern MPEG – 2 System – Transport Stream [10] to the elementary stream multiplexing. To understand what the elementary streams are, it is necessary to understand how the digital sign construction is done. In the first place the audio captured by the microphone and the video captured by the camera are sent separately to the audio codifier and to the video codifier. The stream of bits codifiers created separately is called elementary stream. Once they are multiplexed in an only stream of bits, the elementary stream is entitled transport stream. Two kinds of data structures can be multiplexed in a transport stream: the Packetized Elementary Stream-PES and the sections. The sections are structures defined to transport the tables that are not known as PSI-Program Specific Information. The ABNT NBR 15603[11] specifies in details the structure to build the basic information related to the PSI which are part of the Brazilian system of terrestrial digital television.

For the recommendation system proposed in this paper, the SDT table transports the name of the broadcasting station and the name of the service. The Brazilian system digital TV allows a broadcasting station to transmit more than one service (in the analogical system, known as channel) while the EIT table is responsible for the transportation of the name of the program, start time, duration and complementary information in its descriptors. For example, the descriptor of extended events of the EIT table allows the service provider (broadcasting station) to specify a summary of the program. These tables together transport essential information to present the EPG and they are very important in our recommendation system.

IV. SYSTEM OVERVIEW

The recommendation system proposed in this paper is based on Ginga middleware where the procedural applications are developed using Java^{TM} language and declarative applications in NCL. As mentioned before, the version used was the open source version of Ginga-NCL middleware.

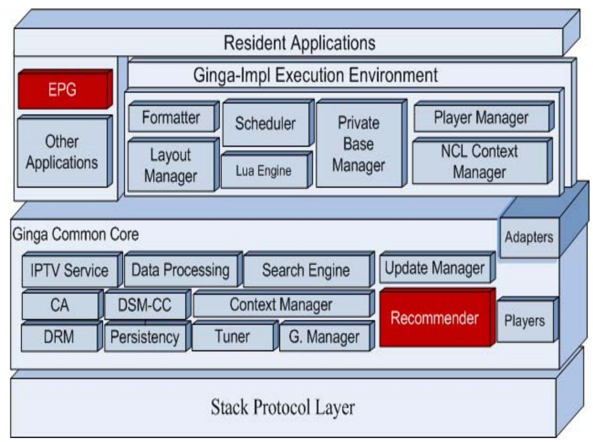

Figure 1. Ginga Middleware Architecture (adjusted with the recommendation system)

Fig. 1 presents its architecture consisting of three layers:

- Resident applications responsible for the exhibition (frequently called presentation layer).
- Ginga Common Core, a set of modules responsible for the data processing, information filtering in the transport stream, data stability.
- It is the architecture core; Stack protocol layer responsible for supporting many communication protocols like HTTP, RTP and TS.

The proposed system extends the Ginga middleware functionalities including new services in the Ginga Common Core layer.

The Recommender module is the main part of the recommendation system and it is inserted in the Common Core layer of Ginga-NCL architecture. The Recommender module is divided in two parts. The first one describes the components integrated to the source code of the middleware such as Agent Local, Agent Schedule, Agent Filter and Agent Data. The second part describes the components added to the set-top Box: Sqlite [12], a C library which implements an attached SQL database and Weka (Waikato Environment for Knowledge

Analysis) [13], open source code which provides a set of algorithms to learn about the machine and the data mining.

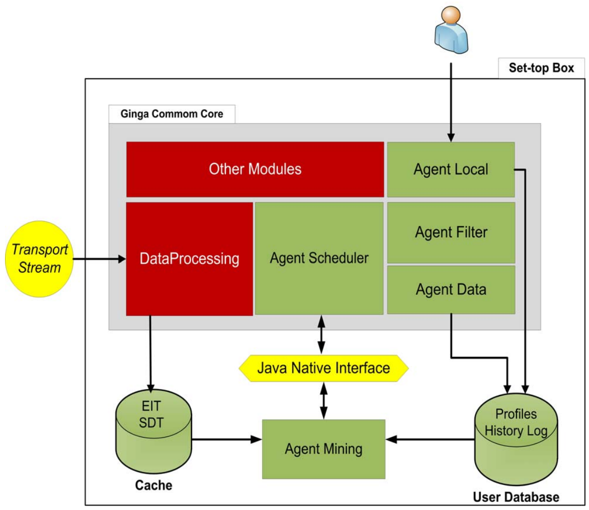

Figure 2. Recommender Module Architecture

Fig. 2 presents the Recommender module architecture.

A. *Implemented Modules*

This subsection describes the modules added to the Ginga middleware source code and the extensions implemented to provide a better connection between middleware and the recommendation system.

Agent Local is the module responsible for constant monitoring of the remote control. Any interaction between the viewer and the control is detected and stored in the database. The Agent Local is essential for the recommendation system that uses implicit approach to realize the profile.

Agent Scheduler is the module responsible for periodically request the data mining. Data mining is a process that demands time and processing, making its execution impracticable every time the viewer requests a recommendation. Agent Scheduler module guarantees a new processing every 24 hours preferably at night, when the set-top Box is in standby.

Agent Mining uses the algorithm package provided by Weka to realize the data mining. Agent Mining module accesses the information in the viewer's behavior background and the programming data from the EIT and SDT tables stored in cache to realize the data mining.

Agent Filter & Agent Data The raw data returned by the Agent Mining module need to be filtered and later stored in the viewer's database. The Agent Filter and Agent Data modules are responsible for this function. The Agent Filter module receives the data from the mining provided by the Agent Mining and eliminates any information that is not important keeping only those which are relevant to the recommendation system such as the name of the program, time, date, service provider and the name of the service. The Agent Data module receives the recommendations and stores them in the viewer's database.

B. *Data Mining Algorithms*

In order to define which mining algorithm implement in the Mining Agent module, tests with three algorithms were performed: C.45, Naïve Bayesian and Apriori. The algorithms tests were performed using the data set provided by IBOPE.Mídia The IBOPE Mídia is the company of the IBOPE group responsible for the communication, media, consumption and audience research. The IBOPE Mídia is already know by its researches in the audience area, but operates also in the advertisement investment area and in quantitative researches in all kinds of communication channels, whether it is Television, Radio, publishing and alternative media.

The data are related to 8 families with different social-economic profiles. The visualization behavior was collected during 4 weeks, minute-by-minute in each house.

In order to choose the algorithm, two STBs characteristics were considered: the quantity of memory and the processing capacity.

The C.45 is a classification algorithm based on decision trees using the share and conquer concept. The Naïve Bayes is a classifier based on statistics, it is fast and efficient when applied to a big data set and is similar in performance to the classifiers based on decision trees. The last compared algorithm is Apriori. It is an association algorithm applied to discover patterns hidden in the data set. The algorithm seeks for affinity among items and expresses it in the way of rules, for example, "70% of the visualization time on Mondays between 7:00 p.m. and 8:00 p.m. is news". Another efficient algorithm is the SVM (Support vector machines). For the proposal of this paper, SVM was not used due to limitations imposed by the STB hardware. Next, we present the results of the algorithms comparison considering processing speed and recommendation accuracy. The accuracy is calculated using the following formula:

$$\mathbf{Ef = (\alpha / \beta)\, 100} \qquad \mathbf{(1)}$$

Where Ef corresponds to the system efficacy and varies from 0 to 100, α is the number of recommendations viewed by the TV viewers and β is the number of recommendations performed. Table 1 presents the results obtained after the analysis of the background of 8 families during 5 weeks.

TABLE I. COMPARISON AMONG ALGORITHMS

Algorithm	Average Time (s)	Accuracy Biggest Value Obtained among 8 houses
C.45	65	71.22 %
Naïve Bayes	54	70.90 %
Apriori	62	72.10 %

The conclusion is that the three algorithms have similar performance, however, with the great quantity of data analyzed, around 43 thousand tuples, Apriori algorithm had a better performance in processing time and accuracy. In such case, we chose to include it as a classification algorithm in the *Mining Agent* module.

V. IMPLEMENTATION

This section presents implementation details of the recommendation system. The great challenge in the implementation of this system was extend the Ginga-NCL middleware functionalities avoiding alterations in the original source code. The loose coupling between Ginga-NCL middleware classes and the extension codes proposed in this paper is essential to keep the compatibility with future middleware versions. The extensions realized in Ginga-NCL middleware source code are presented below.

A. Extensions in Ginga-NCL

The Ginga-NCL middleware source code is composed by many classes written in C/C++ language and distributed among five main modules: ginga-cpp, gingancl-cpp, ncl30, telemidia-link-cpp and telemidia-util-cpp. In order to implement the recommendation system, the functionality of gingacc-cpp and gingancl-cpp modules was extended.

The gingacc-cpp module is responsible for managing the reception of the transport stream, the demultiplexing and the decodification. The decodification process has three stages: identification of the type of table, extraction of the table and the storage in a volatile memory. Ginga-NCL middleware does not implement mechanism to store in cache the EIT and SDT tables transmitted by service providers. To include this functionality, a new class was implemented. This new class can manage the storage in cache. From new implemented services the Ginga-NCL middleware started to provide such essential services for this recommendation system.

The following components were implemented and included as a Ginga Common Core extension: Agent Scheduler module manages the Agent Mining module and guarantees data mining every 24 hours preferably at night, when the set-top Box is in standby. Agent Scheduler module is implemented in C/C++ language. To have access to the Weka package data mining functions written in Java^{TM} language, it was necessary to use JNI (Java Native Interface), a set of APIs (Application Program Interface) which allows the communication between C/C++ language and Java^{TM} language. Agent Scheduler module was implemented as a Thread daemon. With the purpose of saving unnecessary processing resources, Agent Scheduler remains in sleep mode and check in regular periods of time the set-top Box intern clock to decide the right moment to request the mining algorithm execution to the Agent Mining.

Agent Scheduler module was developed to solve a developing problem detected in data mining phase. Data mining demands a lot of processing and time and its execution is not practicable every time the TV viewer requests a recommendation. In order to solve this performance problem, *RecommenderTV* system executes the mining every 24 hours and stores the outcome in the viewer's database. When the TV viewer requests a recommendation, the *RecommenderTV* check set-top Box internal clock to get the time and the date and then search in the database what programs are recommended considering the current date and time. This approaching guarantees the efficacy of the system because all recommendations are stored in the database in advance. In previous versions of *RecommenderTV* system, data mining was processed every time the TV viewer requested a recommendation. This approaching revealed itself as extremely ineffective.

Agent Filter module waits Agent Mining execute its algorithm. As soon as Agent Mining concludes the data mining process, a message is sent to Agent Filter module which performs a parser on the raw data generated by the mining process. When the processing is done Agent Filter request agent Data module services to insert in the database the program recommendation which will be available to the TV viewer in the next 24 hours.

B. Modules Integrated to Ginga-NCL

In order to fulfill some necessities of the recommendation system, two modules were integrated to Ginga-NCL middleware. This subsection describes the Sqlite database and the Weka data mining package integration.

Sqlite is a library which implements a SQL database and it is written in C language. Due to the fact that it is written in C and it provides the entire source code, its integration with Ginga-NCL middleware written in C/C++ language was easier. In order to perform this integration, it was necessary to link the sqlite libraries with the middleware libraries. The sqlite database was chosen due to three facilities: 1) it is written in C language; 2) it was projected to operate in attached devices; 3) it allows Weka mining module to access the information stored in the viewer's database.

The Weka mining package is composed by a set of algorithms which implement different techniques of data mining. Its integration with Ginga-NCL middleware was divided in two very different stages. The first stage defines how Ginga-NCL middleware written in C/C++ and Weka written in Java^{TM} communication is realized. The second stage gives details on how the Weka module communication was implemented with the database that provides the TV viewer's behavior background and the access to EIT and SDT tables proceeding from the providers. Weka is written in Java^{TM} language and its integration with Ginga-NCL middleware written in C/C++ needed a bridge to allow the communication between the different languages. To make possible the integration between Weka and Ginga, it was necessary to use the resources provided by API Java^{TM} Native Interface. API allowed Agent Scheduler module to access **Apriori** mining algorithm provided by Weka. This algorithm looks for affinity among the items and expresses it through standards like "70% of visualization time on Mondays around 7 and 8 p.m. It is news".

In order to the mining algorithms implemented by Weka can produce program recommendations, it is necessary two sets of well planned data: the TV viewer's behavior background and the program grade available to the TV viewers by the providers through EIT and SDT tables. The details on how Weka module accesses this information are described below. The TV viewer's behavior background is built by collecting and storing any interaction between the viewer and the set-top Box. Table 2 presents a sample of the behavior background information stored in database used to show the outcomes presented in this paper.

TABLE II. VIEWER GROUP VISUALIZATION BACKGROUND

channel	Program Name	category of the program	Day	Time	Min
5	P1	News	Monday	Night	8
2	P2	News	Tuesday	Night	20
8	P3	Kids	Tuesday	Morning	40

In order to process the data mining, the Mining module has direct access to the database and recovers the TV viewer's behavior background. From the point of view of the system performance, this communication between mining module and sqlite database is important. Without this communication, it would be necessary to implement a new module responsible for recover the database information and then make such data available to the mining algorithm. The second data set necessary to make possible the data mining is the program guide. The program guide is composed by information sent by providers through EIT and SDT tables. These tables are stored in cache and are available to be recovered and processed by the Mining module. Ginga-NCL Middleware does not implement storage mechanism in cache of EIT and SDT tables. This functionality was implemented by the TV Recommender system. Agent Mining implements a service which performs the data in cache reading and converts at the same time to the format interpreted and processed by Weka package.

VI. EXPERIENCES

A. Results

This subsection describes the results after a month of interaction with the recommendation system. In order to measure the evolution of the recommendation offered to the TV viewer, the following formula was applied:

$$\mathbf{Ef = (\alpha / \beta)\ 100} \qquad (2)$$

Where EF is the efficacy of the recommendation system, ranging from 0 to 100, α is the recommendation number accepted by the TV viewers and β is the number of recommendation presented. In order to monitor these indicators, two control variables of the viewer's data base were preserved. β variable is increased every time the TV viewer requests a recommendation, α variable is only increased when the TV viewer accept some of the suggestions offered by the recommendation system. Thus, the efficacy of the system can be calculated daily, only verifying the variables values and calculating the efficacy of the system.

Fig. 3 shows the recommendation evolution in five weeks using the *RecommenderTV* system. In the end of each week, the data with information about the number of recommendation requested and accepted were taken out of the viewer's database and applied in formula (2). The little efficacy of the recommendation system in the first week is explained when the data base is analyzed with the TV viewer interactions with the set-top Box. Little information was available to the data mining algorithms. On the second week, an evolution in the recommendation system was noted. As the TV viewer interaction with the set-top Box was stored, the amount of information to the mining algorithm increased and there was an improvement in the quality of recommendation. On the third week, there was not an increase in the efficacy as meaningful as it was on the first and second week. The fourth and fifth week presented a stability of the system, keeping a success efficacy of about 80%.

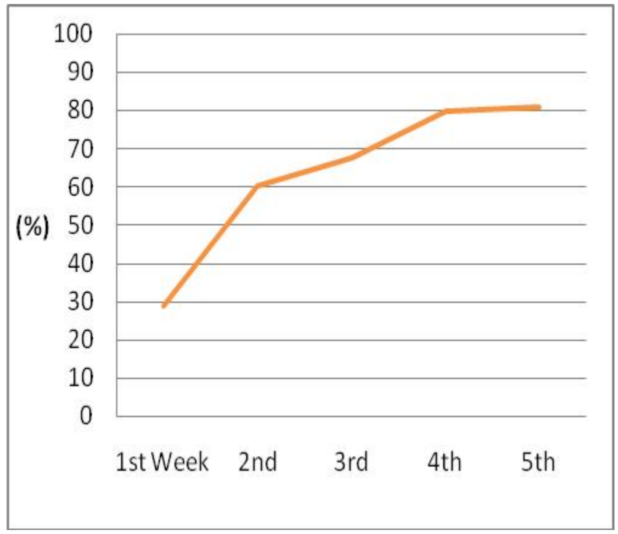

Figure 3. Recommender System

The superiority of a system which uses data mining mechanisms in comparison with random systems and does not use any method to recommend programs is evident. This superiority can be proved in Figure 4 chart. Only in the first week, when the *RecommenderTV* system had little information, the efficacy percentages were close.

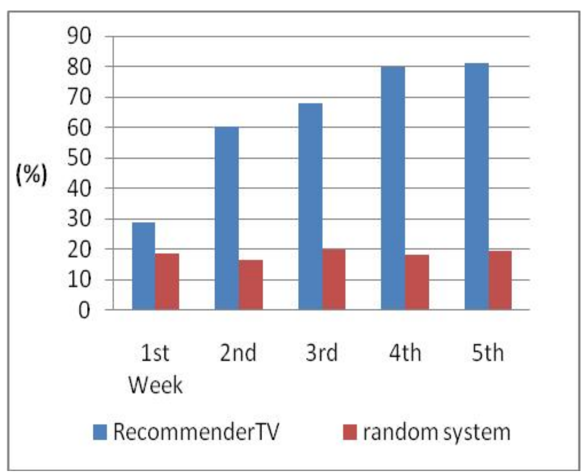

Figure 4. comparison between Recommender TV & Random System

Fig. 5 shows *RecommenderTV* system. The application used as front-end is written in NCL and allows the TV viewer to search the recommendation list selecting the wanted program.

Figure 5. RecommenderTV System (Front-End)

VII. CONCLUSION

With the appearance of digital TV, a variety of new services (in the analogical system, channels) will be available. This information overload requires the implementation of new mechanisms to offer facilities to the viewers looking for their favorite programs. These new mechanisms suggesting the viewers programs are known as recommendation systems.

All the extensions to Ginga middleware proposed in this paper as the inclusion of new classes in the Common Core, the Sqlite database linking and the Weka mining module aim the increasing of the efficacy in the recommendation system and the inclusion of new functionalities non-existent yet in Ginga.

With the purpose of simulating the life cycle of the digital television Brazilian system which starts in the service provider making the audio, video and data available and finishes in the viewer set-top Box, a simulation environment was implemented. This environment is composed by three service providers transmitting audio, video and data to Ginga-NCL Virtual set-top Box which is in accordance with the Brazilian standards executing in a MiniPc. Despite Brazil broadcasts digital sign for more than one year, this is limited to audio and video; data and consequently sections and tables, and it is not yet a reality in the country. It was a problem in the validation of the recommending system because it is necessary to access two important tables: EIT and SDT and these tables are not broadcasted by the service providers in Brazil so far. In face of this limitation, an EIT and SDT table generator was implemented in accordance with the Brazilian standards.

In order to validate the recommendation system, it would be adequate to store EIT and SDT tables in the set-top Box in advance, but for a simulation closer to a real situation such tables were sent in a transport stream what allowed verifying data demultiplexing and decoding as well as testing new modules included in Ginga-NCL middleware in order to store the data decoded in cache.

With the implementation, it was clear that without the alterations proposed in this paper, a recommendation system implementation is impracticable. The necessity to keep the viewer's behavior information in data base associated with the necessity of storing the information coming from the service providers require the linking of new modules to Ginga middleware and the extension of others. This paper described the complete implementation of a recommendation system compatible with Ginga middleware. The expectation for future research is to extend the functionalities implemented in *RecommenderTV* system, allowing the interoperability with other devices through UPnP^{TM}/DLNA protocol in a home networks.

ACKNOWLEDGMENT

We would like to thank the FAPESP for support in the project TIDIA-AE (process 05/60653-1) and FINEP for support in the project "**Avaliação do middleware Ginga**" (01.07.0110.00).

REFERENCES.

[1] H. Zhang, S. Zheng and J. Yuan. 2005, "A Personalized TV Guide System Compliant with MHP", IEEE Transactions on Consumer Electronics, pages 731-737, Vol. 51, No. 2, MAY 2005.

[2] L.F.G. Soares, R.F. Rodrigues, M.F. Moreno. 2007, "Ginga-NCL: The declarative Environment of the Brazilian Digital TV System", Journal of the Brazilian Computer Society. V.12, n.4, p.37-46, March 2007.

[3] Souza Filho, G. L., Leite, L. E. C., Batista, C. E. C. F. 2007, "Ginga-J: The Procedural Middleware for the Brazilian Digital TV System.", Journal of the Brazilian Computer Society, v. 12, n. 4, p. 47-56, March 2007.

[4] Ginga-NCL Virtual STB, (March 2009), Available at: http://www.ncl.org.br/ferramentas/index_30.html.

[5] J. Xu, L.Zhang, H. Lu and Y. Li. 2002, "The Development and Prospecto f Personalized TV Program", Proceedings of the IEEE 4th Internacional Symposium on Multimedia Software Engineering (MSE'02), California, USA, 2002.

[6] P.Cotter and B. Smyth. 2000, "PTV: Intelligent Personalized TV Guides", Proceedings of the 12th Innovative Applications of Artificial Intelligence Conference (IAAI 2000), Austin, Texas, 2000.

[7] L. Kurupati, S.Gutta, D. Schaffer, J. Martino and J.Zimmerman. 2001, "A multi-agent TV recommender", Proceedings of the International Conference on User Modeling 2001, Workshop on Personalization in Future TV, Sonthofen, Germany, 2001.

[8] L.Ardisson, F.Portis, P. Torasso. 2001, "Architecture of a system for the Generation of Personalized Eletronic Program Guides", Proceedings of the International Conference on User Modeling 2001, Workshop on Personalization in Future TV, Sonthofen, Germany, 2001.

[9] S. H. Hsu, M. H. Wen, H. C. Lin, C. C. Lee, C. H. Lee. 2007, "AIMED – A personalized TV Recommendation System", Proc 2007 Interactive TV: A Shared Experience. 5th European Conference, EuroITV 2007, Amsterdam, the Netherlands.

[10] ISO/IEC 13818-1.2008. Information technology – Generic coding of moving pictures and associated audio information - Part 1: Systems.

[11] ABNT, (March 2009) Available at: http://www.abnt.org.br/tvdigital/abnt_pt.html.

[12] Sqlite, (March 2009) Available at: http://www.sqlite.org/.

[13] Weka,(March 2009) Available at: http://www.cs.waikato.ac.nz/ml/weka/.

[14] Han,J., Kamber,M. 2006. "Data Mining: Concepts and Techniques", Morgan Kaufmann,2006.